Training-Free Layout Control with Cross-Attention Guidance
Supplementary Material

Minghao Chen Iro Laina Andrea Vedaldi
Visual Geometry Group, University of Oxford
{minghao, iro, vedaldi}@robots.ox.ac.uk

This supplementary material contains the following parts:

- **Implementation Details.** We provide more details of the experimental settings, including the network architecture and noise scheduler.
- **Evaluation Dataset and Metrics.** We provide the details of dataset and evaluation metrics used in the experiments part.
- **Ablation Study.** A detailed quantitative evaluation is presented to understand the impact of various components and hyper-parameter selections. We investigate the influence of guided steps, layer-specific losses, and the loss scale factor for backward guidance.
- **Analysis on Initial Noise.** We demonstrate that different prompts with the same initial noise generate images with similar layouts. Therefore, a good choice of initial noise is essential for the success of guidance. Additionally, we quantitatively prove that using the defined loss on cross-attention allows for optimal initial noise selection, enhancing guidance performance.
- **Analysis on Different Tokens.** We visualize the cross-attention map of different prompts and provide extra experiments about controlling the layout of the generated image with only padding tokens.
- **More Examples.** We provide additional examples of our method, including examples under VISOR [3] protocol and real image editing examples.
- **Ethics.** We provide discussion on ethical considerations related to data usage.

1. Implementation Details

We provide additional details of our experimental settings.

Network Architecture. In all experiments, we use the Stable Diffusion (SD) V-1.5 [9] as our base model without any architecture modification. The diffusion model is trained in the latent space of an autoencoder. Specifically, the diffusion model adopts the U-Net [10] architecture with a relative downsampling factor of 8. The downsampling branch of the U-Net has three sequential cross-attention blocks. The mid part of the U-Net has only one cross-attention block. The up-sampling branch of the U-Net has three sequential cross-attention blocks. In each cross-attention block, there are repeated layers following the order: ResBlock → Self-Attention → Cross-Attention. The cross-attention blocks in the down-sampling branch, mid part, and up-sampling have 2, 1, and 3 such repeated patterns, respectively.

Noise Scheduler. The LMSDscheudler is utilized in all of our experiments with 51 time steps and beta values starting at 0.00085 and ending at 0.012, following a linear scheduler. We also adopt class-free guidance, as suggested in [4], with a guidance scale of 7.5, consistent with prior work [9].

2. Evaluation Datasets and Metrics

VISOR [3]. We follow the evaluation process described in [3] to compute the VISOR metric, which is designed to quantify the spatial understanding abilities of text-to-image models. This metric focuses on two-dimensional relationships, such as *left*, *right*, *above*, and *below*, between two objects. We measure object accuracy (OA), which is the probability that the generated image contains both objects specified in the text prompt. $\mathrm{VISOR}_{\mathrm{uncond}}$ is the probability that generating both objects with correct spatial relationship, and $\mathrm{VISOR}_{\mathrm{cond}}$ is the conditional probability of correct spatial relationships being generated, given that both objects were generated correctly. To generate text prompts for evaluation, we use the 80 object categories from the MS COCO dataset [6], resulting in a total of $80 \times 79 \times 4 =$ 25,280 prompts considering any combination of two object categories for each spatial relationship. For each prompt, we generate a single image. As layout guidance inputs we split the image canvas into two, vertically or horizontally, to create two adjacent bounding boxes depending on the type of spatial relationship defined by the text prompt. This only imposes a weak constraint on the layout and can be done automatically (no user intervention is required). For a fair comparison to previous methods that are evaluated in [3], we use the same detection model (OWL-ViT [7]) as in [3] when computing the VISOR metric.

COCO 2014 [6] We randomly sampled 1000 images with their annotations for evaluation from the COCO 2014 validation dataset. The bounding boxes in COCO 2014 are not always grounded in the corresponding caption. Therefore, we append the object labels to the caption as the text prompt for image generation following a similar setting in [1, 2]. Besides, we only pick one to three bounding boxes with areas covering at least 5% of the image panel per sample following the setting in [2]. To assess the quality of the generated images we compute the FID score between the sampled 1000 images from COCO and generated images. We use an open-vocabulary object detector (Detic [12]) to obtain the respective grounding on generated images, which allows quantifying *layout fidelity* using common detection metrics such as average precision (AP). The vocabulary of the detector is constrained to all the COCO object labels.

Flickr30K Entities [8] Finally, we evaluate our method on the Flickr30k Entities dataset [8, 11], which contains image-caption pairs. Since the dataset provides visual groundings of the textual descriptions, we sample a single caption per image and its corresponding bounding boxes and use this as input to perform layout-controlled guidance with SD. We generate a total of 1,000 images using samples from the validation set. Similarly to the metric used in COCO 2014, we compute the FID score between the original images and the generated ones and use AP as a metric of layout control. To enhance the reliability of the detector, we convert each phrase in the Flickr30 dataset into a single noun (*e.g.*, *ball*) and filter out unrelated nouns, resulting in a total of 303 categories. For each image, the target vocabulary for Detic is defined by the grounded entities in the corresponding caption. To avoid contaminating the evaluation process with perceived human attributes (such as gender, age, occupation, etc.), we also convert all instances of people (man, woman, child, boy, girl, policeman, student, etc.) to the super-class "person" in the target vocabulary for Detic. Since then the *person* category is predominant, we calculate average precision separately for this category (AP_p) but also report the mean average precision across all categories (mAP).

3. Ablation Study

In this section, we supplement the ablation studies in the main paper with quantitative evaluations, studying the impact of the guided steps, loss scale factor, and the effect of backward guidance on different layers of the denoising network. We followed the same setting as described above and in Section 4.1 (main paper) using 1000 captions and their corresponding bounding boxes from the Flickr30K Entities [8] dataset to generate images with a pre-specified layout.

Guidance Step	FID (↓)	AP_p (↑)	mAP (↑)	Inference Time
0	76	19.4	8.7	∼ 4sec/image
2	**81.2**	29.7	13.7	∼ 4sec/image
5	81.4	30.3	15.6	∼ 6 sec/image
10	82.0	33.5	**16.7**	∼ 8 sec/image
15	82.3	35.5	14.7	∼ 10 sec/image
20	83.2	35.6	15.3	∼ 12 sec/image
30	83.5	**35.7**	15.3	∼ 15 sec/image

Table A1. Ablation study on guidance steps.

Impact of Guidance Step. Firstly, we explore the effects of guided steps we perform in the diffusion process. The results are shown in Tab. A1, we evaluate image quality (FID), AP_p, layout control (mAP) while varying the number of *guided* steps. We found no improvement in mAP after 10 steps, and FID gradually deteriorates. We hypothesize that this decline may result from potentially shifting the latent vector away from the distribution that corresponds to the original text embedding. Besides, we could see that when increasing the guided steps in the diffusion process, the computation time increases. This is a trade-off question. Generally, a range of 2-10 guidance steps suffices, but users can fine-tune this based on their specific requirements.

Impact of Layers. Secondly, we study the behavior of different layers, by applying backward guidance on the cross-attention maps across different layers of the network. The results are shown in Table A2. As stated in Section 4.4 and illustrated in the table, layers of the down-sampling branch are the least likely to conform to layout control (with Down-1 < Down-2 < Down-3 in terms of mAP). In general, high-resolution blocks (such as Down-1 or Up-3) should not be used to control the layout. To achieve the best trade-off between image quality and layout control, a combination of the mid-block (Mid-1) and the first cross-attention block in up-sampling branch (Up-1) of the U-Net is the optimal choice overall.

Impact of Loss Scale Factor. We follow the same setup to evaluate the scale factor η used as the strength of the loss for backward guidance. In Table A3 we report the FID, AP_p and mAP for different loss scale factors. When the loss scale is set to 5–50, the FID is low compared to a larger loss scale factor, indicating that the quality with a loss scale factor of 5–50 is generally good. To achieve better control over the layout, the loss scale factors of 20–50 have the lowest AP_p and mAP. According to the experiments, a loss scale

Base Model	Down-1	Down-2	Down-3	Mid-1	Up-1	Up-2	Up-3	FID (↓)	AP_p (↑)	mAP (↑)
Stable Diffusion [9]	✓	✓	✓					81.3	31.1	13.2
	✓							83.5	23.1	10.0
		✓						82.0	24.0	10.9
			✓					82.2	34.5	14.2
				✓				82.1	30.0	15.2
				✓	✓			82.0	33.5	**16.7**
				✓		✓		86.3	30.9	14.0
				✓			✓	84.1	23.5	10.5
					✓	✓	✓	84.5	35.6	16.5
					✓			**81.2**	**36.0**	15.1
						✓		87.5	35.0	14.3
							✓	85.0	25.6	9.8

Table A2. Ablation study of loss constraints on different layers.

Loss Scale	FID (↓)	AP_p (↑)	mAP (↑)
5	82.5	28.3	12.4
10	82.0	30.0	14.5
20	**81.1**	34.7	15.4
30	82.0	33.5	**16.7**
50	83.8	**35.8**	15.6
100	88.4	34.9	14.3
200	99.2	32.2	13.8
500	129.7	26.2	9.2

Table A3. Ablation study of the loss scale factor.

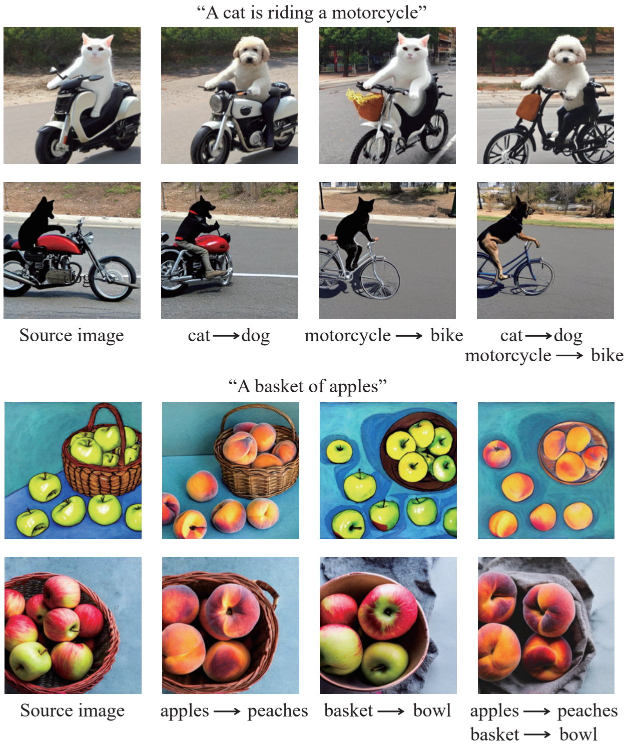

Figure A1. Each row has the same initial noise. We could see that even if we changed the object word in one sentence, the overall layout remains similar.

factor of 20–50 works generally well. This factor can be adjusted by the user to get more realistic images or achieve better control over the layout.

4. Analysis on Initial Noise

We conduct an in-depth analysis of the effects of initial noise. As illustrated in Figure A1, the initial noise reveals significant spatial information about the layout. Notably, altering sentence words does not affect this final layout significantly. Figure A2 offers a visual comparison of scenarios with and without noise selection. The results indicate that our backward guidance achieves better control when noise selection is employed. Furthermore, Table A4 quantitatively assesses the impact of noise selection on COCO 2014 and Flickr30K datasets. Methods incorporating noise selection consistently outperform others, underscoring the efficacy of our loss as a noise selection metric.

5. Analysis on Different Tokens

Next, we study the type of information carried by different tokens and their corresponding cross-attention maps, which is relevant for layout guidance.

Removing Word Tokens. We first show that the *padding* tokens convey a significant amount of semantic information. In Figure A3, we randomly pick a subset of captions from MSCOCO [6] and generate images using the Stable Diffusion model and the full caption as the input prompt. As a comparison, after the captions pass through the text encoder, we replace the token embeddings of each caption with the embeddings of its corresponding padding tokens, thus creating a prompt that consists only of padding tokens. Then, we use this prompt to generate images. Surprisingly, despite only generating from padding (*i.e.*, non-word) token embeddings, we observe that the generated images (Word Drop in Figure A3) closely follow both the semantics and the layout of the image generated from the

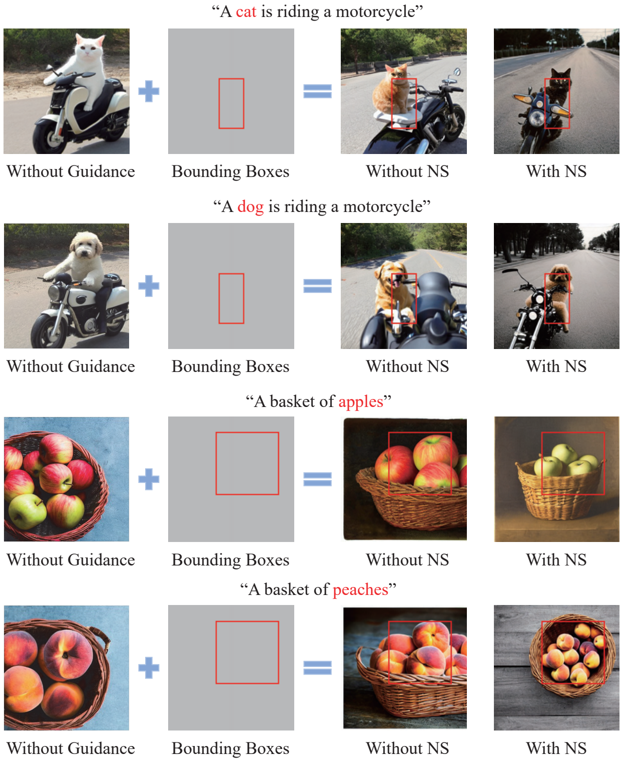

Figure A2. We qualitatively compare the generated results with and without noise selection (NS). The results show that with noise selection, our backward guidance achieves better layout control.

Base Model	NS	COCO 2014		Flickr30K		
		FID (↓)	mAP (↑)	FID (↓)	mAP (↑)	AP_P (↑)
Stable Diffusion	✗	74.4	33.6	82.0	33.5	16.7
Stable Diffusion	✓	**73.3**	**35.7**	**78.9**	**35.6**	**17.9**

Table A4. Ablation Study on Noise Selection (NS).

full-text prompt. Thus, the figure clearly demonstrates that the padding tokens contain the information of the whole sentence. This further justifies why in forward guidance padding tokens cannot be ignored, *i.e.*, it would be insufficient to attempt to control selected word tokens only (main paper, Figure 5). In backward guidance, however, controlling the cross-attention maps of padding tokens is not necessary; this is now done by back-propagating and updating the latent, which subsequently changes the cross-attention maps of all tokens, even those that are not explicitly controlled.

Cross-Attention Maps of Special Tokens. During our experiments, we found that the cross-attention of the padding tokens has a strong connection to the foreground of the generated images. We illustrated this in Figure 4 (main paper), which shows that the cross-attention maps of padding tokens resemble saliency maps, while the cross-attention maps of the start tokens are mostly complementary to those of padding tokens (*i.e.*, they capture what can be considered as background). In Figure A4, we show more examples of the cross-attention maps of the *start* and *padding* tokens. The captions are randomly taken from MSCOCO [5]. This figure further highlights the observation that cross-attention maps of these special tokens contain important semantic and spatial information. For example, in the first row, given "A short train traveling through a mountainous landscape" as the input prompt, the cross-attention map of the padding tokens aligns with the generated train and the start token focuses on the background of the generated image.

Layout Control with Only Padding Tokens. Motivated by the examples above, we perform backward guidance only on the cross-attention maps of padding tokens to control the spatial layout of all foreground objects simultaneously (as a group). Some examples are shown in Figure A5. This figure verifies our assumption that by guiding the cross-attention map of the padding tokens alone one can control the composition of the images at the foreground/background level.

6. More Examples.

More Examples under VISOR Protocol. We show more examples under the VISOR protocol in Figure A6 and Figure A7. Our method generates the correct spatial relationships as shown in the figures. There are also some failure cases, such as the last row in Figure A6. Our method fails to generate both a fork and a carrot. This is an inherited problem from the Stable Diffusion model. However, in most cases, layout guidance helps generate *all* entities in the text prompt, even when the unguided Stable Diffusion fails (*e.g.*, as is often the case with atypical scene compositions), as well as conforming to a specific spatial arrangement.

More Image Editing Examples. We show more examples of real image editing in Figure A8. Specifically, we train for 500 steps to learn the embedding of $\langle * \rangle$ with text inversion and then 150 steps fine-tuning of the text encoder and denoiser network with Dreambooth. After finalizing the model, we perform inference with our backward guidance using different text prompts and user-specified bounding boxes. As shown in the figure, we manage to change the context, layout, and style of the given real image.

7. Ethics

We use the Flick30K Entities and MS-COCO datasets in a manner compatible with their terms. Some of these images may accidentally contain faces or other personal information, but we do not make use of these images or image regions. For further details on ethics, data protection, and copyright please see https://www.robots.ox.ac.uk/~vedaldi/research/union/ethics.html.

References

[1] Omri Avrahami, Thomas Hayes, Oran Gafni, Sonal Gupta, Yaniv Taigman, Devi Parikh, Dani Lischinski, Ohad Fried, and Xi Yin. Spatext: Spatio-textual representation for controllable image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18370–18380, 2023. 2

[2] Guillaume Couairon, Marlène Careil, Matthieu Cord, Stéphane Lathuilière, and Jakob Verbeek. Zero-shot spatial layout conditioning for text-to-image diffusion models. *arXiv preprint arXiv:2306.13754*, 2023. 2

[3] Tejas Gokhale, Hamid Palangi, Besmira Nushi, Vibhav Vineet, Eric Horvitz, Ece Kamar, Chitta Baral, and Yezhou Yang. Benchmarking spatial relationships in text-to-image generation. *arXiv preprint arXiv:2212.10015*, 2022. 1, 2, 9, 10

[4] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 1

[5] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. In *Proc. ECCV*, 2014. 4

[6] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2014. 1, 2, 3

[7] Matthias Minderer, Alexey Gritsenko, Austin Stone, Maxim Neumann, Dirk Weissenborn, Alexey Dosovitskiy, Aravindh Mahendran, Anurag Arnab, Mostafa Dehghani, Zhuoran Shen, et al. Simple open-vocabulary object detection with vision transformers. *arXiv preprint arXiv:2205.06230*, 2022. 2

[8] Bryan A. Plummer, Liwei Wang, Christopher M. Cervantes, Juan C. Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. *International Journal of Computer Vision (IJCV)*, 2017. 2

[9] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10684–10695, 2022. 1, 3

[10] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015. 1

[11] Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *TACL*, 2014. 2

[12] Xingyi Zhou, Rohit Girdhar, Armand Joulin, Philipp Krähenbühl, and Ishan Misra. Detecting twenty-thousand classes using image-level supervision. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022. 2

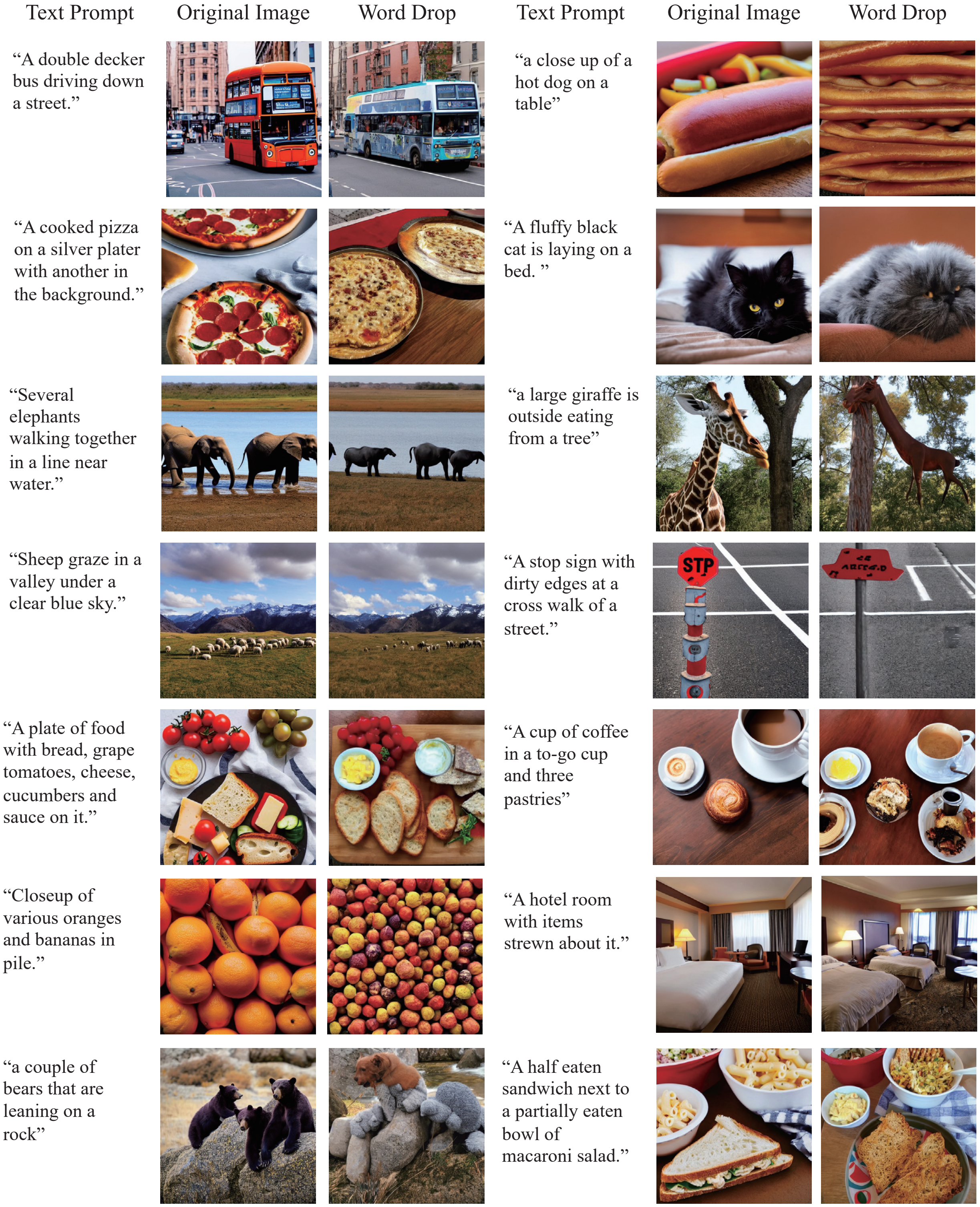
Text Prompt
Original Image
Word Drop
Text Prompt
Original Image
Word Drop
“A double decker bus driving down a street.”
“a close up of a hot dog on a table”
“A cooked pizza on a silver plater with another in the background.”
“A fluffy black cat is laying on a bed. ”
“Several elephants walking together in a line near water.”
“a large giraffe is outside eating from a tree”
“Sheep graze in a valley under a clear blue sky.”
“A stop sign with dirty edges at a cross walk of a street.”
STP
“A plate of food with bread, grape tomatoes, cheese, cucumbers and sauce on it.”
“A cup of coffee in a to-go cup and three pastries”
“Closeup of various oranges and bananas in pile.”
“A hotel room with items strewn about it.”
“a couple of bears that are leaning on a rock”
“A half eaten sandwich next to a partially eaten bowl of macaroni salad.”

Figure A3. Generating images without “seeing” the full-text prompt. We replace the token embeddings for all words in each caption with their *padding* token embeddings (word drop). We observe that the generated images after word dropping exhibit similar semantics and layout to the images generated from the full-text prompt, suggesting that significant information about the image is contained in padding tokens.

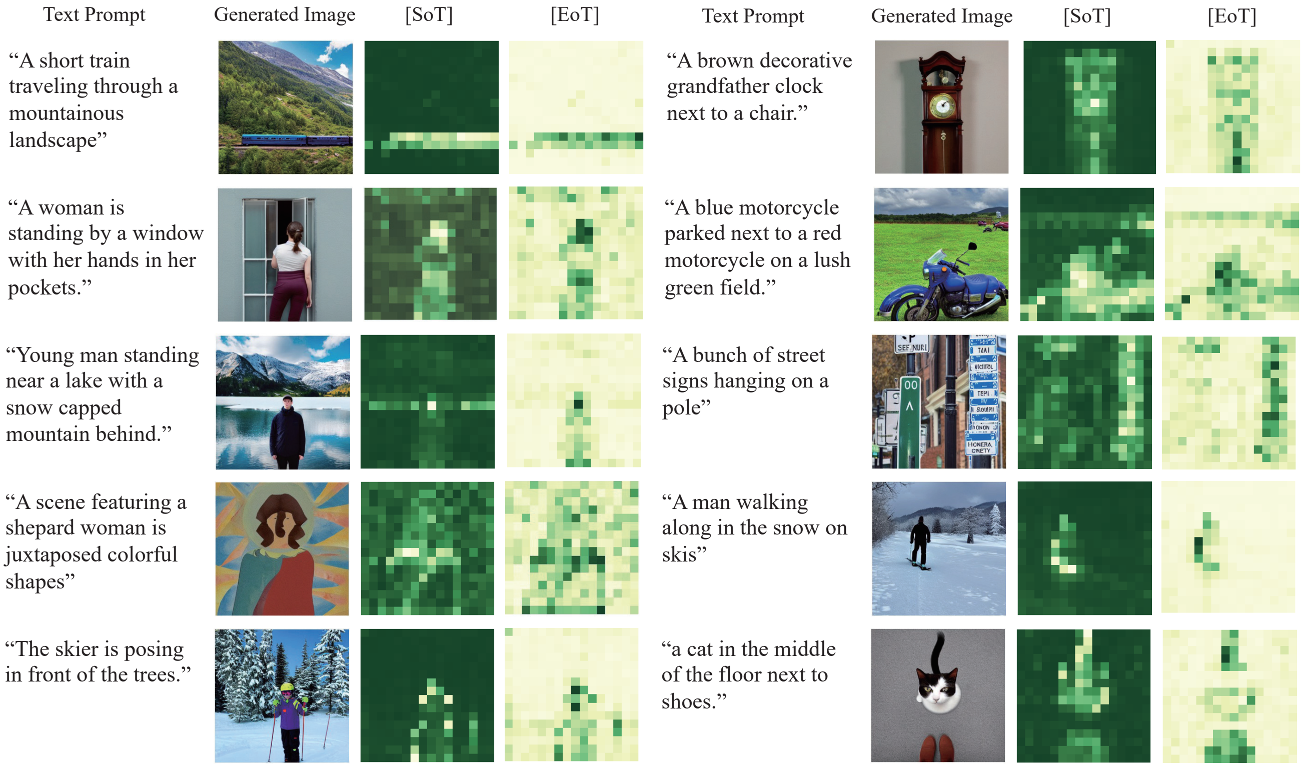
Text Prompt
Generated Image
[SoT]
[EoT]
Text Prompt
Generated Image
[SoT]
[EoT]
“A short train traveling through a mountainous landscape”
“A brown decorative grandfather clock next to a chair.”
“A woman is standing by a window with her hands in her pockets.”
“A blue motorcycle parked next to a red motorcycle on a lush green field.”
“Young man standing near a lake with a snow capped mountain behind.”
“A bunch of street signs hanging on a pole”
“A scene featuring a shepard woman is juxtaposed colorful shapes”
“A man walking along in the snow on skis”
“The skier is posing in front of the trees.”
“a cat in the middle of the floor next to shoes.”

Figure A4. Visualization of cross-attention maps of start token (`[SoT]`) and padding tokens (`[EoT]`) at the final step of inference. Cross-attention maps are taken from the first cross-attention block of the up-sampling branch of U-Net and averaged over all attention heads.

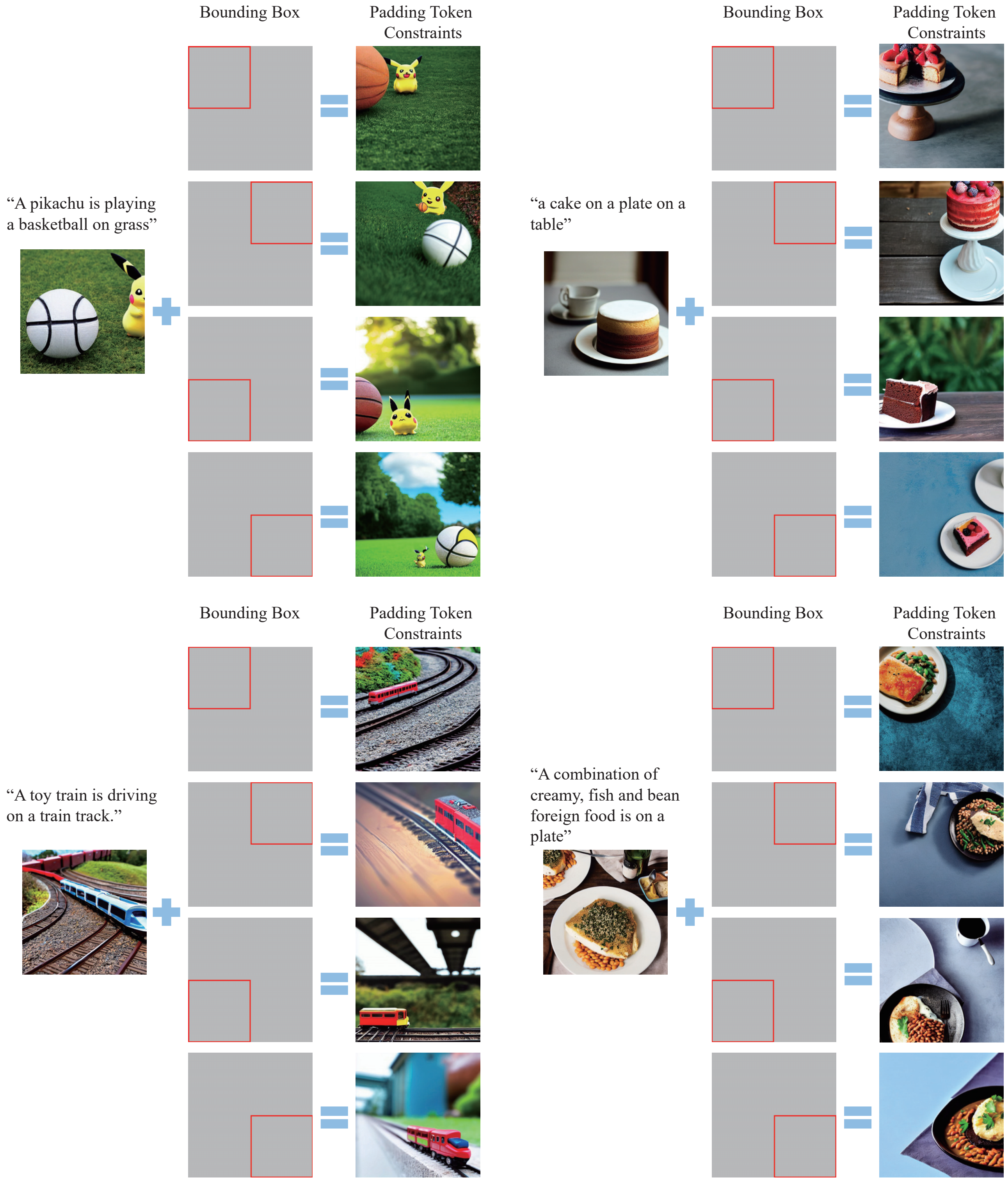
Bounding Box
Padding Token Constraints
"A pikachu is playing a basketball on grass"
Bounding Box
Padding Token Constraints
"a cake on a plate on a table"
Bounding Box
Padding Token Constraints
"A toy train is driving on a train track."
Bounding Box
Padding Token Constraints
"A combination of creamy, fish and bean foreign food is on a plate"

Figure A5. Backward guidance *only* on the *padding* tokens. We observe that the cross-attention of padding tokens typically represents the foreground of the generated image. Therefore, by spatially guiding the cross-attention maps that correspond to padding tokens, we can control the position of the foreground, which may include multiple objects (e.g., "pikachu" and "basketball").

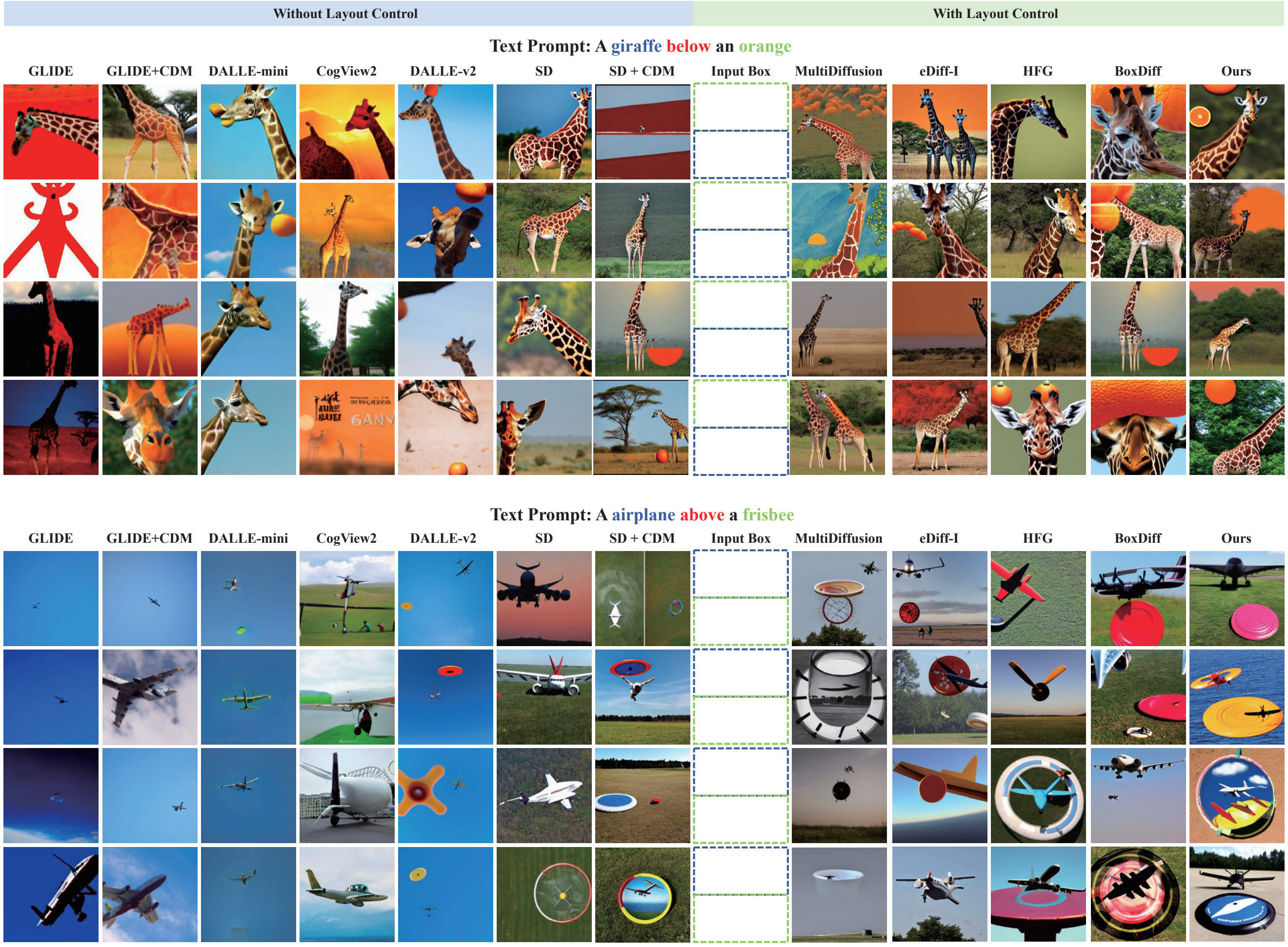

Figure A6. Qualitative comparison between different generative models. For each prompt, we generate four images. Some images of other models are from the demo website of [3].

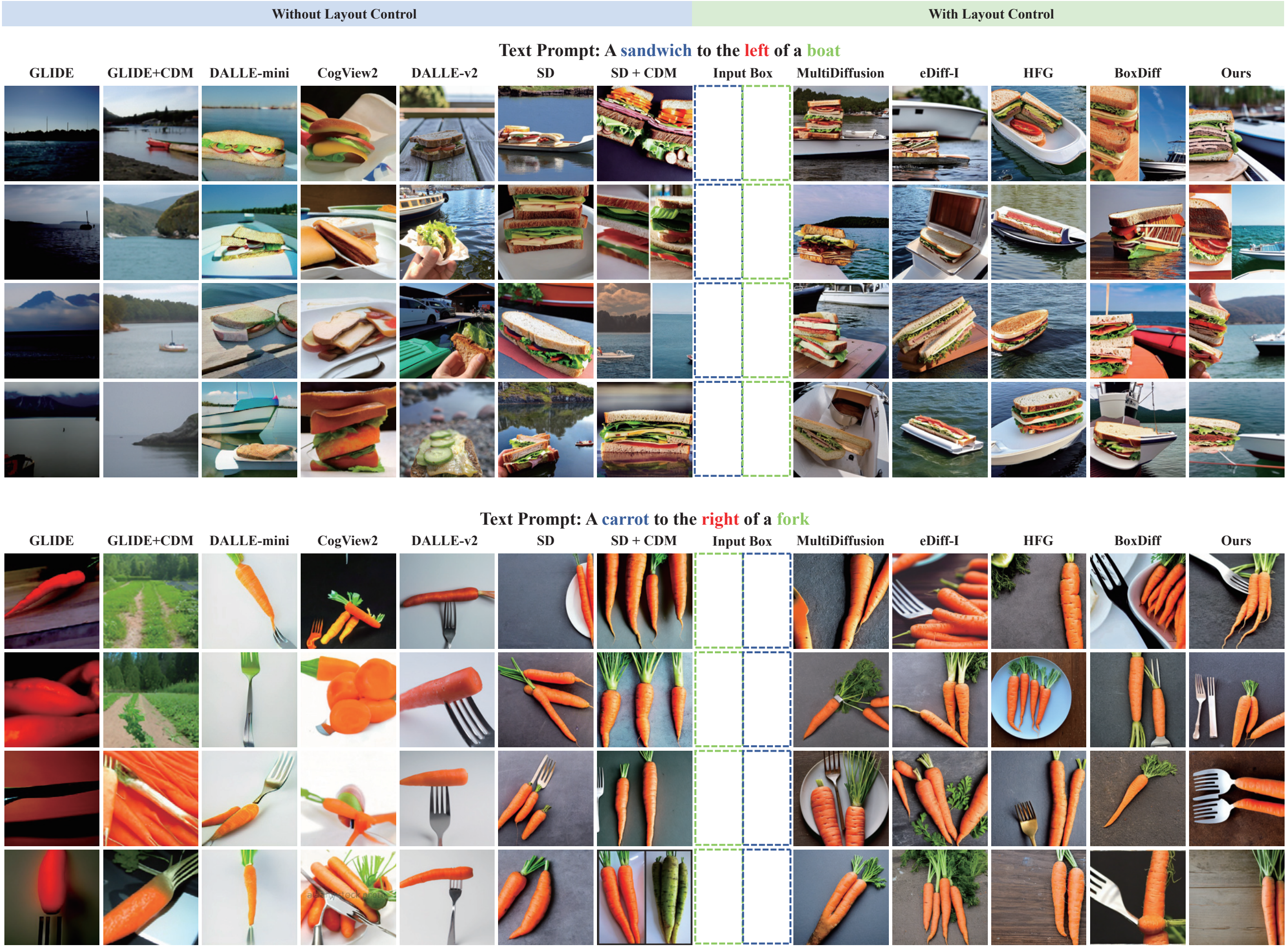

Figure A7. Qualitative comparison between different generative models. For each prompt, we generate four images. Some images of other models are from the demo website of [3].

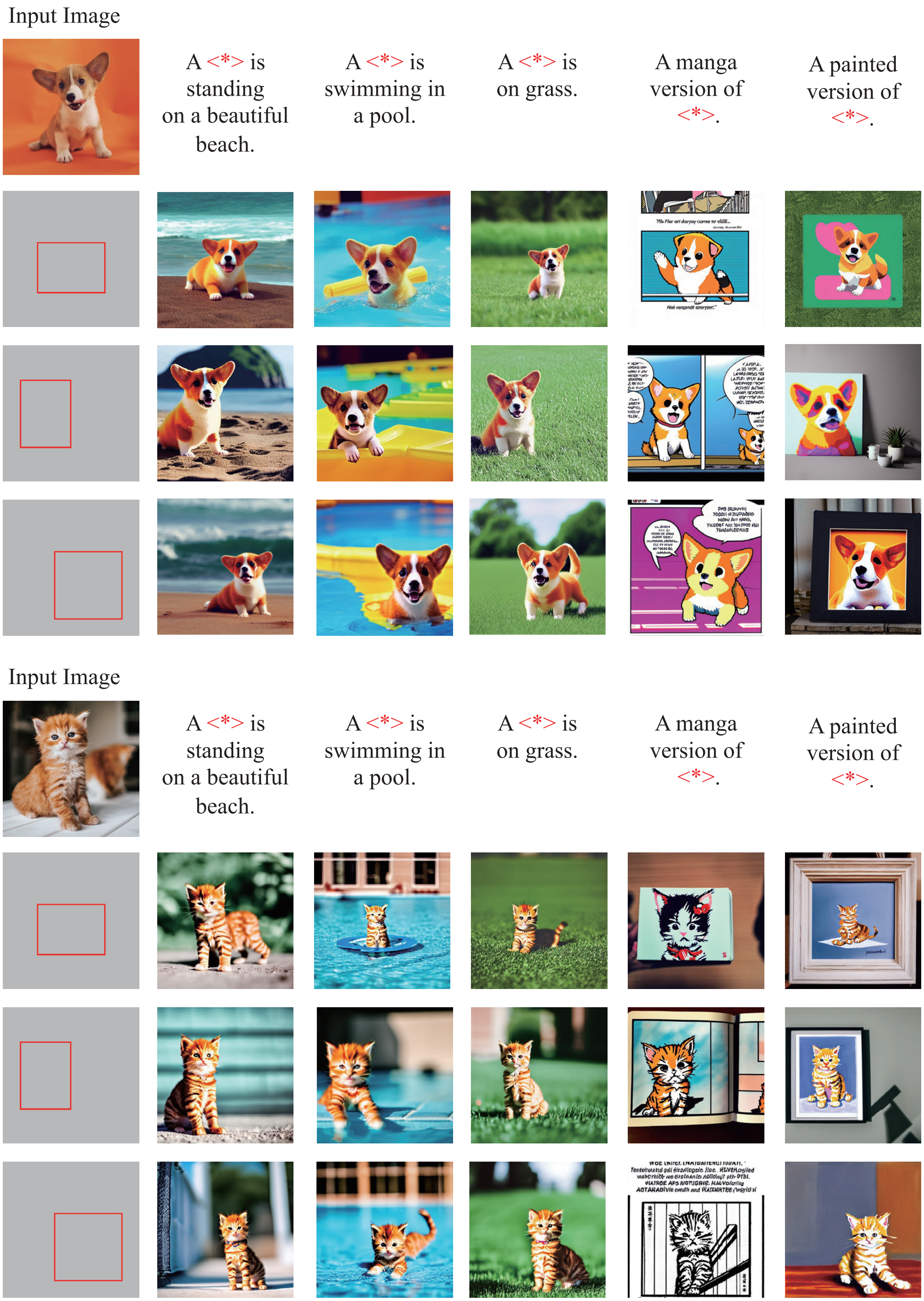
Input Image
A <*> is standing on a beautiful beach.
A <*> is swimming in a pool.
A <*> is on grass.
A manga version of <*>.
A painted version of <*>.
Input Image
A <*> is standing on a beautiful beach.
A <*> is swimming in a pool.
A <*> is on grass.
A manga version of <*>.
A painted version of <*>.

Figure A8. More examples of real image editing. ⟨∗⟩ is the learned token that encodes the object in the real image.